



IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of : **Confirmation No. 3053**
Carsten BURMEISTER et al. : Docket No. 2001_0944A
Serial No. 09/909,823 : Group Art Unit 2664
Filed July 23, 2001 : Examiner Fox, Jamal A.
METHOD AND APPARATUS FOR : **Mail Stop Amendment**
TRANSMITTING DATA PACKETS

CLAIM OF PRIORITY UNDER 35 USC 119

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

THE COMMISSIONER IS AUTHORIZED
TO CHARGE ANY DEFICIENCY IN THE
FEES FOR THIS PAPER TO DEPOSIT
ACCOUNT NO. 23-0975

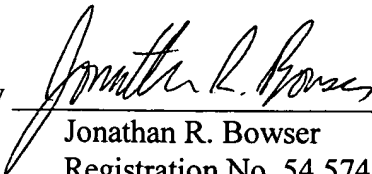
Sir:

Applicants in the above-entitled application hereby claim the date of priority under the International Convention of European Patent Application No. 00119566.8, filed September 7, 2000, as acknowledged in the Declaration of this application.

A certified copy of said European Patent Application is submitted herewith.

Respectfully submitted,

Carsten BURMEISTER et al.

By 

Jonathan R. Bowser
Registration No. 54,574
Attorney for Applicants

JRB/nrj
Washington, D.C. 20006-1021
Telephone (202) 721-8200
Facsimile (202) 721-8250
August 26, 2005

This Page Blank (uspto)



**Europäisches
Patentamt**

**European
Patent Office**

**Office européen
des brevets**

Bescheinigung

Certificate

Attestation

Die angehefteten Unterlagen stimmen mit der ursprünglich eingereichten Fassung der auf dem nächsten Blatt bezeichneten europäischen Patentanmeldung überein.

The attached documents are exact copies of the European patent application described on the following page, as originally filed.

Les documents fixés à cette attestation sont conformes à la version initialement déposée de la demande de brevet européen spécifiée à la page suivante.

Patentanmeldung Nr. Patent application No. Demande de brevet n°

00119566.8

Der Präsident des Europäischen Patentamts;
Im Auftrag

For the President of the European Patent Office

Le Président de l'Office européen des brevets
p.o.

MÜNCHEN, DEN
MUNICH,
MUNICH, LE

24/05/05

This Page Blank (uspto)



Anmeldung Nr:
Application no.: 00119566.8
Demande no:

Anmeldetag:
Date of filing: 07.09.00
Date de dépôt:

Anmelder/Applicant(s)/Demandeur(s):

MATSUSHITA ELECTRIC INDUSTRIAL CO., LTD.
1006, Ohaza Kadoma
Kadoma-shi,
Osaka 571-8501
JAPON

Bezeichnung der Erfindung/Title of the invention/Titre de l'invention:
(Falls die Bezeichnung der Erfindung nicht angegeben ist, siehe Beschreibung.
If no title is shown please refer to the description.
Si aucun titre n'est indiqué se referer à la description.)

Method and apparatus for transmitting data packets

In Anspruch genommene Priorität(en) / Priority(ies) claimed /Priorité(s)
revendiquée(s)

Staat/Tag/Aktenzeichen/State/Date/File no./Pays/Date/Numéro de dépôt:

/00.00.00/

Internationale Patentklassifikation/International Patent Classification/
Classification internationale des brevets:

H04L/

Am Anmeldetag benannte Vertragstaaten/Contracting states designated at date of
filing/Etats contractants désignées lors du dépôt:

AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU MC NL PT SE

This Page Blank (uspto)

07. Sep. 2000

METHOD AND APPARATUS FOR TRANSMITTING DATA PACKETS

The invention generally relates to a method and apparatus for transmitting data packets in a packet stream over an unreliable channel, and in particular to
5 transmitting data packets having compressed headers.

Several communication technologies exist for transmitting data from one terminal to another terminal. The most commonly used techniques are cellular telephony and the Internet. Further developments are media-on-demand and conversational services such as Internet telephony. Most of these services require the transport of
10 real-time data including audio and video contents.

The Real-time Transport Protocol (RTP) provides means for such purposes. RTP is an Internet protocol for transmitting data real-time or nearly real-time. RTP itself does not guarantee real-time delivery of data but it does provide mechanisms for the sending and receiving applications to support streaming data. Typically, RTP runs
15 on top of the UDP protocol. UDP (User Datagram Protocol) is a connectionless protocol that, like TCP, runs on top of IP networks. Unlike TCP/IP, UDP/IP provides no error recovery services, but instead offers a direct way to send and receive datagrams over IP network.

While RTP has been developed for fixed networks, it may nevertheless be used in
20 mobile networks. However, one problem in using RTP over mobile networks is the limited bandwidth in the mobile channel. This is because each of the protocols RTP, UDP and IP has its own header. A packet will then, in addition to link layer framing, have an IP header of 20 bytes, a UDP header of 8 bytes, and an RTP header of 12 bytes, thus summing up to at least 40 bytes.

25 This header is highly redundant, and to decrease the amount of overhead, header compression mechanisms have been developed. Header compression protocols remove the redundancy of the header and encode the information in an efficient way. This may lead to a compression of the original header down to one byte in the best case.

A system using a header compression protocol is illustrated in FIG. 1. The transmitter comprises a compressor 100 which is used for compressing the original header. The compressed header is then transmitted to the receiver and is there decompressed by the decompressor 110.

5 The context 120 is the state which the compressor uses to compress the header. The context is a set of variables and consists basically of an uncompressed version of the header fields of the last header. Besides the actual header fields, the context comprises additional variables, such as first order differences of header fields that have been detected to be constant for a series of successive packets. The context
10 can also contain additional information describing the packet stream, for example the typical inter-packet increase in sequence numbers or timestamps.

In operation, the compressor 100 and the decompressor 110 are required to maintain a common context. When the context 130 of the decompressor 110 is not consistent with the context 120 of the compressor 100, header decompression will
15 fail. This situation can occur when data packets are transmitted over unreliable, e.g. wireless, channels because packets may then be lost or damaged between compressor 100 and decompressor 110.

It is therefore necessary to initiate a resynchronization procedure once the context 130 of the decompressor 110 has become invalid. For this purpose, update (UP)
20 packets are provided for transmitting information contained in the context 120 of the compressor 100, to the decompressor 110. Thus, by using UP packets, the context 130 is updated.

The performance of a header compression scheme can be described with two parameters, compression efficiency and robustness. A robust scheme tolerates
25 errors on the link over which header compression takes place without losing additional packets, introducing additional errors or using more bandwidth. Using UP packets increases on the one hand the robustness, but decreases compression efficiency, since UP packets are large in size. Therefore, in addition to UP packet, non-update (NUP) packets are used which are very small and which only depend on
30 the previous UP packet. Thus, NUP packets do not update the context so that, if a NUP packet gets lost, the context 130 of the decompressor 110 continues to be valid, so that the receiver is still able to decompress the following packets.

The packet stream to be compressed usually behaves very regularly. Most of the header fields are constant and do not change during the life-time of the stream. Some fields do change with each packet (e.g. sequence number and timestamp). If the values of these fields are synchronized to the sequence number and thus can be calculated from this number the stream is regular. Irregularities in these fields disturb this synchronization, e.g. because of a non-linear jump in the RTP-timestamp field. With an irregularity it is not possible to calculate the values of the changed fields from the sequence number. These irregularities might occur quite frequently, e.g. on the average every second for a conversational audio stream.

- 10 In case an irregular change has occurred, information about it has to be transmitted to the decompressor. Therefore, either UP or NUP packets have to be extended by this information. This can for instance be done by setting an extension bit in the header and by placing the irregularity information into a predefined extension field of the header. However, using extending UP (extUP) packets decreases the
15 robustness while using extended NUP (extNUP) packets decreases the compression efficiency.

It is therefore the object of the invention to provide a method and apparatus for transmitting data packets in a packet stream, capable of improving both the efficiency and robustness.

- 20 This object is solved by the invention as defined in the independent claims.

- According to the invention, a decision is made whether to send extUP or extNUP packets, based on at least one packet stream parameter. Thus, the invention allows for determining the optimum conditions for both, compression efficiency and packet stream robustness by dynamically adapting the transmission scheme to the channel
25 and packet stream properties. This reduces the mean header size even when irregular changes of the packet stream occur.

- The invention is in particular advantageous as it allows for sending extNUP packets in case the irregular change is only valid for a short number of packets. This is because if in case of a short irregularity extUP packets were used the
30 decompressor's context would be easily invalidated and the decompressor would not be able to decompress all subsequent packets until a new UP packet is received

correctly. That is, the invention increases the packet stream robustness compared with a transmission scheme in which irregularities are transmitted in extUP packets only.

Moreover, the invention is on the other hand advantageous because it refrains from only using extNUP packets. As the number of NUP packets is usually greater than the number of UP packets, by sending extUP packets wherever possible the invention allows for increasing the compression efficiency.

Preferred embodiments of the invention are defined in the dependent claims.

The invention will now be described with reference to the accompanying drawings in which:

FIG. 1 illustrates a compressor/decompressor system, in which UP and NUP packets are used;

FIG. 2 is a flow chart illustrating the process of deciding when to transmit extUP or extNUP packets according to the invention;

FIG. 3 is a flow chart illustrating the process of estimating the maximum number of consecutive packet loss according to a preferred embodiment of the invention; and

FIGs. 4a and 4b are flow charts illustrating preferred embodiments of the process of estimating the number of packets for which the irregular change is valid.

Preferred embodiments of the invention will be described in more detail hereinafter.

As will be apparent from the discussion below, the invention makes use of at least one packet stream parameter. Packet stream parameter means any channel, packet stream and compressor-state property which can at least indirectly provide some information which might be suitable for deciding when and how to send information about an irregular change to the decompressor. In the preferred embodiments, the following parameters are used:

N_1 : the number of packets that have been sent since the last update sequence;

N_2 : the maximum number of consecutive packet loss over the channel, i.e. the maximum number of consecutively lost packets in the packet stream; and

N_3 : the number of subsequent packets of the stream for which an irregular change is valid, i.e. the time length of an irregularity in units of data packets.

Referring now to FIG. 2, in deciding when to use extUP and when to use extNUP packets the compressor 100 first determines in step 200 whether an irregular change has occurred. If no irregular change has occurred there is no need to transmit extended packets at all, and the process returns. If it is however determined in step 200 that an irregular change has occurred the compressor 100 checks two separate conditions for deciding which packets to extend.

In checking the first condition, the compressor 100 obtains the parameter N_1 in step 210. Then, the parameter N_2 is retrieved in step 220. The parameter N_2 may for instance have previously been estimated using the process which is described below in the context of FIG. 3. The compressor then can simply retrieve the parameter from a storage unit or any other kind of data buffer.

Once the parameters N_1 and N_2 have been obtained, the compressor 100 performs a comparison between these values in step 230. If the parameter N_1 is not greater than the parameter N_2 it is decided to transmit extNUP packets in step 270. Otherwise the process continues with step 240.

In checking the second condition, the compressor retrieves in step 240 the parameter N_3 , again by accessing previously estimated values. It is then determined in step 250 whether N_2 exceeds N_3 , and if so, it is again decided to transmit extNUP packets to the decompressor. Otherwise, the decompressor will in step 260 receive information about the irregular change via extUP packets.

Thus, extended UP packets are transmitted only if both conditions 230, 250 are fulfilled. If at least one condition is not met it is decided to transmit extNUP packets.

By this process, compression efficiency is increased because the irregular change is not transmitted in all packets, i.e. no larger extNUP packets have to be transmitted after the new context is established. Further, by sending extUP packets whenever necessary, robustness is increased.

While in discussing FIG. 2 it has been described that condition 230 is checked before condition 250 it will be appreciated by those of ordinary skill in the art that alternatively, condition 250 may be checked first.

Preferably, the number of extUP packets in one sequence is adapted in step 260 to the parameter N_2 , for reliably establishing the irregularities in the decompressor's context. In a preferred embodiment, the number of extUP packets is set to be equal to N_2 .

As mentioned above, the parameters N_2 and N_3 are preferably retrieved in steps 220 and 240 from any kind of storage unit, and these parameters have to be previously estimated. While FIG. 3 illustrates a preferred embodiment of estimating the parameter N_2 , the generation of N_3 estimates will be described in the context of FIGs. 4a and 4b.

Referring now to FIG. 3, the estimation of the maximum number of consecutive packet loss is based on non-acknowledgement (NACK) packets sent from the decompressor 110 to the compressor 100. NACK packets are sent if an invalid context has been detected by the decompressor due to a UP packet loss. The invalid context is detected upon reception of the first NUP packet which contains a sequence indication bit unequal to the one stored in the decompressor's context.

In step 300, the compressor receives a NACK packet or message from the decompressor and extracts the sequence number of the last correctly decompressed packet, i.e. where the context was still valid, from this NACK message (step 310). Then, the compressor obtains the current sequence number in step 320. From the extracted and the current sequence numbers the compressor is able to calculate the amount of packets sent to the decompressor between the transmission time of the last correctly received packet and the reception time of the NACK message. In step 330, the compressor obtains the Round Trip Time (RTT) which is in this case the time required to trigger and receive the NACK message. Then, the compressor subtracts the RTT value from the calculated amount of packets, thus calculating the number of packets which were lost consecutively (step 340). This number is then made accessible to the compressor as the parameter N_2 .

The estimation of N_3 is preferably be done as depicted in FIGs. 4a and 4b. While in the process of FIG. 4a knowledge about the used codec and its properties is used, the process of FIG. 4b includes observing the packet stream and gaining estimations for the future from the experiences of the past. It will be appreciated by those of
5 ordinary skill in the art, that the processes of FIGs. 4a and 4b may be used alternatively as well as in combination.

In FIG. 4a, the compressor knows the properties of different streams coming from different codecs. This information can be stored in a look-up table of the compressor. In step 400 the compressor checks the RTP Payload Type field of the
10 header to know which is the codec used. Then, the compressor retrieves the necessary information about the codec from the look-up table in step 410 and calculates the parameter N_3 using the retrieved information (step 420).

In the process of FIG. 4b, the compressor retrieves in step 440 observed packet stream properties such as the maximum, minimum, average, variance of the
15 average, etc, of the number of packets for which an irregular change is valid. These properties are preferably stored in a memory of the compressor. From this information, the compressor calculates in step 450 an estimation of the parameter N_3 which depends on the degree of robustness one would like to have. Wishing a higher robustness implies that the chosen value is to be near the minimum number
20 of packets.

As apparent from FIGs. 3, 4a and 4b, the estimation processes further include a step 350, 430 of applying a safety factor. This is to take into account that the calculated values of parameters N_2 and N_3 are estimates only. Thus, in order to ensure the robustness of the scheme, the estimated N_3 is preferably be divided by the safety
25 factor which is greater than one, while the estimated N_2 is preferably be multiplied with this factor.

This Page Blank (uspto)

CLAIMS

1. Method of transmitting data packets in a packet stream, the data packet having compressed headers, the method comprising the steps of:

compressing a header using a context (120);

- 5 transmitting at least one update (UP) packet containing data indicating said context; and

transmitting at least one non-update (NUP) packet;

wherein the method further comprises the steps of:

detecting (300) an irregular change of the packet stream;

- 10 obtaining (310, 320) at least one packet stream parameter (N_2 , N_3); and

transmitting (340, 350) either an extended update (extUP) packet or an extended non-update (extNUP) packet dependent on the determined packet stream parameter, the extended packet including information about the irregular change.

- 15 2. The method according to claim 1, wherein the packet stream parameter is the maximum number (N_2) of consecutive packet loss.

3. The method according to claim 2, further comprising the step of:

20 entering (230) a context update phase if the number of packets (N_1) sent since the last update phase is greater than the maximum number (N_2) of consecutive packet loss.

4. The method according to claim 2 or 3, wherein the maximum number (N_2) of consecutive packet loss has been estimated by extracting (300, 310) a sequence number from a received NACK message and comparing (320, 340) the extracted sequence number with the current sequence number.

- 25 5. The method according to one of claims 2 to 4, wherein the number of extended update packets is set dependent on the packet stream parameter.

6. The method according to one of claims 2 to 5, wherein the step of obtaining at least one packet stream parameter includes obtaining the number (N_3) of subsequent packets for which the irregular change is valid.
- 5 7. The method according to claim 6, further comprising the step of comparing (330) the maximum number (N_2) of consecutive loss and the number (N_3) of subsequent packets for which the irregular change is valid, wherein extended update packets are transmitted (350) only if the number (N_3) of subsequent packets for which the irregular change is valid is greater than the maximum number (N_2) of consecutive packet loss.
- 10 8. The method according to claim 6 or 7, wherein the number (N_3) of subsequent packets for which the irregular change is valid has been estimated by checking (400) the RTP Payload Type field and accessing a codec look-up table.
9. The method according to claims 6 or 7, wherein the number (N_3) of subsequent packets for which the irregular change is valid has been estimated by retrieving
15 (440) observed packet stream properties.
10. The method according to one of claims 1 to 9, wherein the step of obtaining at least one packet stream parameter includes the step of applying (350, 430) a safety factor.
11. Apparatus for transmitting data packets in a packet stream, the data packets
20 having compressed headers, the apparatus comprising:

a compressor (100) for compressing a header using a context (120);

transmission means for transmitting a least one update (UP) packet containing data indicating said context and at least one non-update (NUP) packet;

detection means for detecting an irregular change of the packet stream; and

25 control means for obtaining at least one packet stream parameter and controlling said transmission means to transmit either an extended update (extUP) packet or an extend non-update (extNUP) packet dependent on the determined packet stream parameter, the extended packet including information about the irregular change.

12. Apparatus according to claim 11, arranged for performing the method according to one of claims 1 to 10.

This Page Blank (uspio)

07. Sep. 2000

ABSTRACT

The invention relates to a method and apparatus for transmitting data packets in a packet stream wherein the data packets have compressed headers. Update packets containing data indicating the context used in compressing the headers, and non-
5 update packets are transmitted. According to the invention, an irregular change of the packet stream is detected and at least one packet stream parameter is obtained. Dependent on the determined packet stream parameter, either an extended update packet or an extended non-update packet is transmitted where the extended packets include information about the irregular change. The invention further includes
10 estimating the packet stream parameter and applying a safety factor.

This Page Blank (uspio)

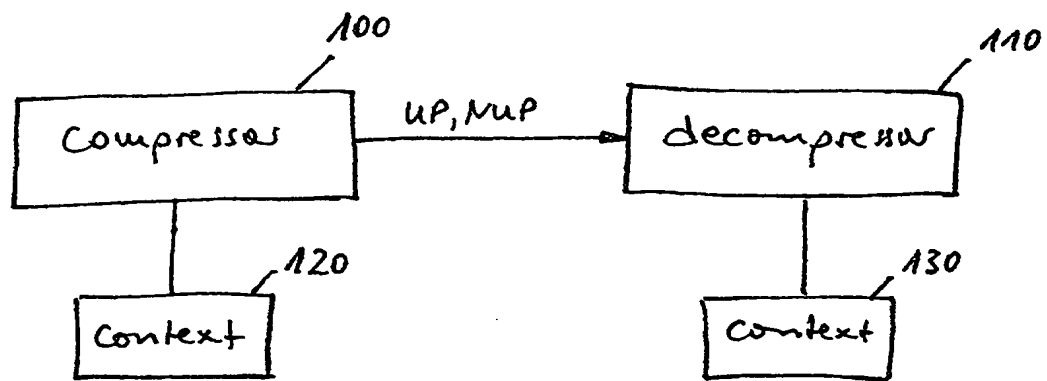


Fig. 1

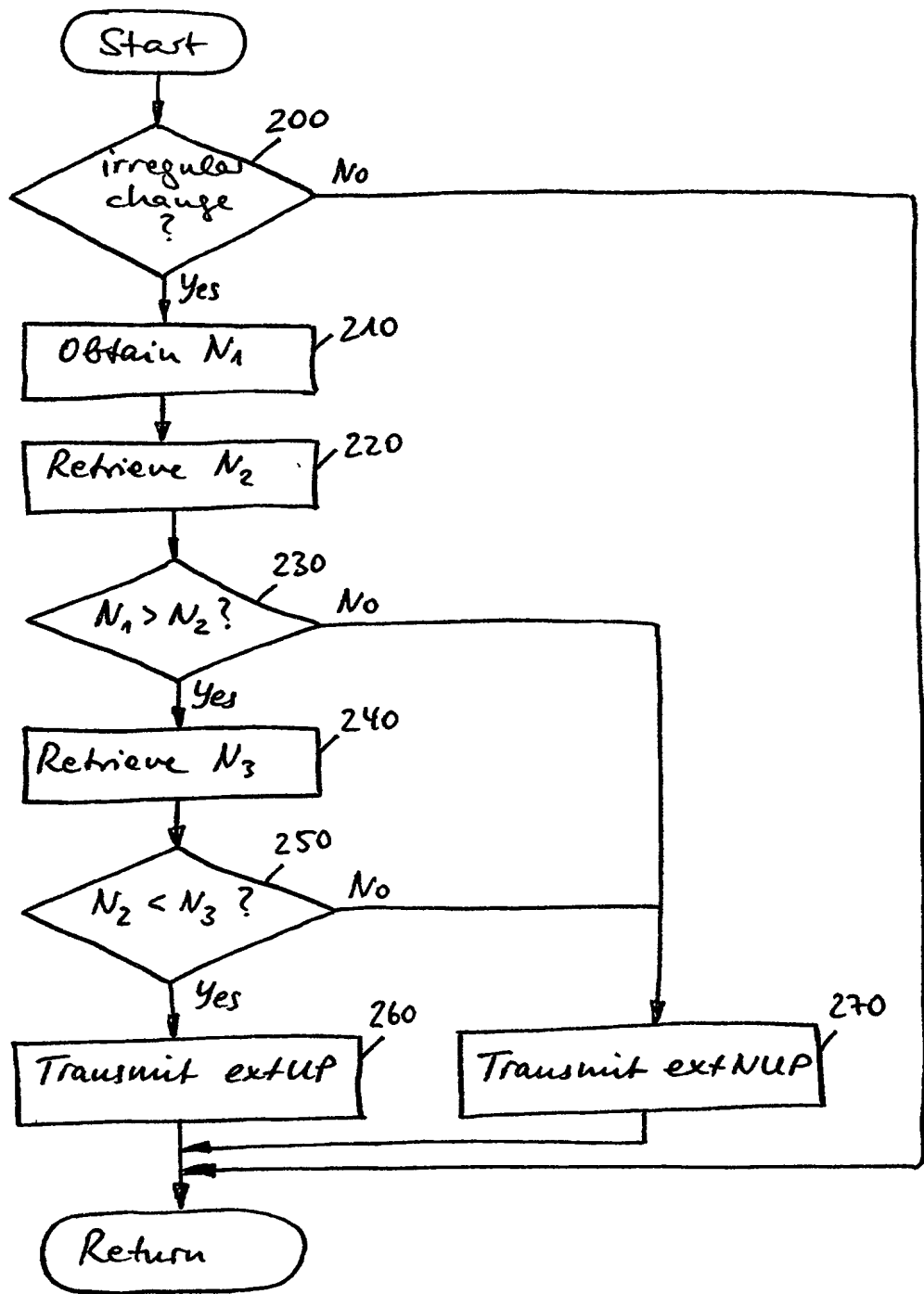


Fig. 2

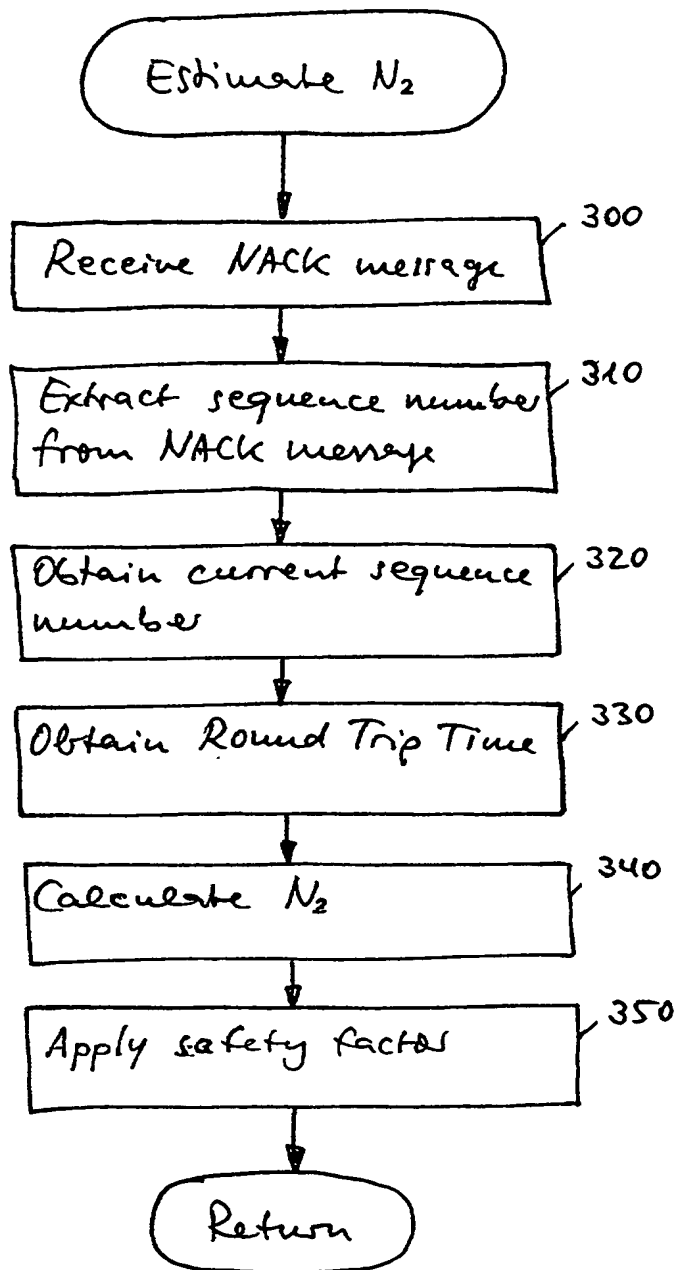


Fig. 3

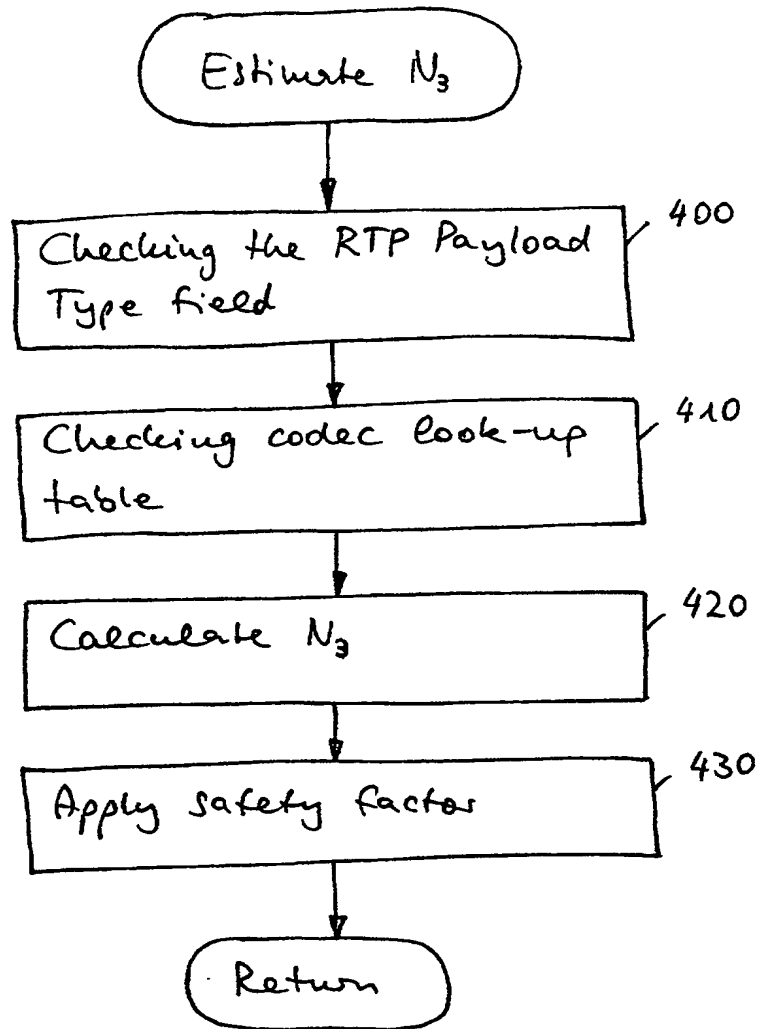


Fig. 4a

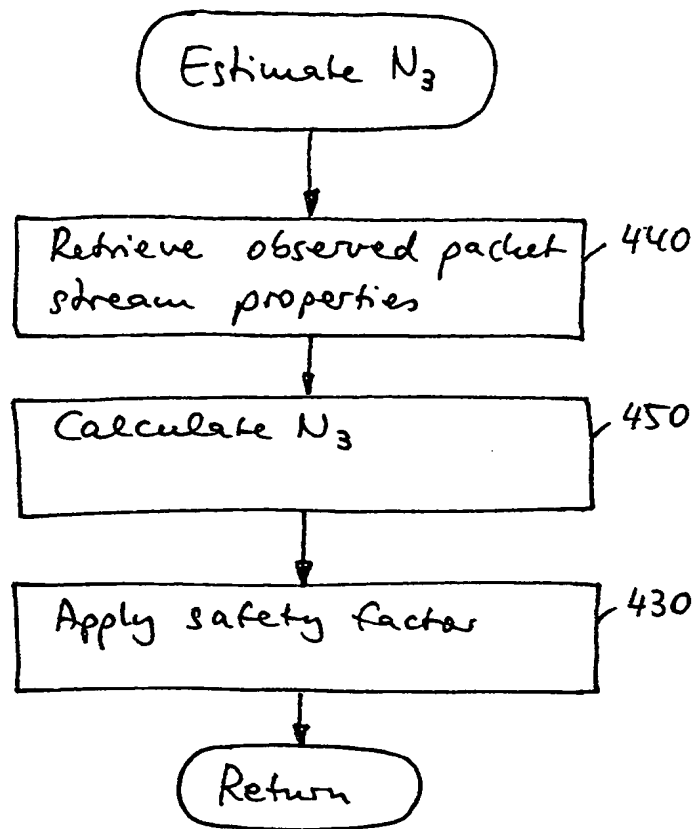


Fig. 4b

This Page Blank (usp10)